9 维护与故障排除

本章将介绍：

——硬件维护

——系统诊断

——CPU 指示灯

——PWR 指示灯

——RUN 指示灯

——CPU 指示灯

——BATT 指示灯

——通讯问题

——I/O 模块的故障排除

——噪音故障排除

——启动系统与程序调试

硬件维护

硬件维护

DL305 是一个低维护系统，只要定期对它做一些检查就能降低许多发生故障的可能性。常规的维护和检查应注意以下两个主要项目：

· 空气质量（环境温度，气流等）

· CPU 电池

空气质量维护

您的系统所处环境的空气质量会影响到系统的性能。如果您的系统是放在控制柜里的，请检查其环境温度是否超过了允许的范围。如果柜子中有空气过滤器，请在必要的时候给予清洗或更换来保证。建议您最好每隔一两个月来对系统来进行一次检查。请保证系统是工作在规定的环境条件下的。

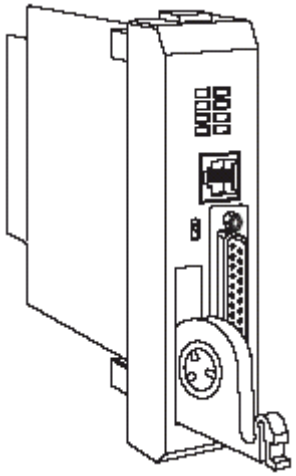
电池报警灯

CPU 有一个电池 LED 来表示其中的电池是否还电量充足。您应该对其进行定期的检查并在必要的时候更换电池。您也可以通过编程来查看电池的电流是否低。SP43 这一特殊继电器在电池需要更换的时候会自动置 ON。

CPU 电池的更换

CPU 电池是用来保持 R 寄存器中的内容及系统参数的。电池使用寿命为 5 年。

注意：在安装或更换 CPU 电池以前，请备份您的 R 寄存器内容及系统参数。您可以通过 DirectSOFT 中保存程序，R 寄存器以及系统参数到个人计算机的硬盘或软盘上。



电池的安装步骤：

1. 按住电池舱门上的塑料片把舱门旋开。
2. 把电池放在纽扣槽中
3. 关上舱门并确保其已可靠到位
4. 把安装日期记录下来

警告：严禁对电池进行充电或把废旧电池扔入火中。因为电池有可能爆炸或产生有毒气体。

可以在编程时使用**系统诊断**

系统诊断

您的 DL305 系统在每个扫描周期都会完成预定义的诊断程序。该诊断功能能够检测到 CPU 和 I/O 模块的各种错误。基本的错误等级有两个，致命错误（fatal）和非致命错误（non-fatal）。

致命错误

致命错误是 CPU 检测到的可能使系统不能安全地或正常地工作的一些错误。如果 CPU 在运行模式下发生了致命错误，CPU 会自动切换到编程模式。（请记住，在编程模式下所有的输出点都会断开。）如果 CPU 在编程模式下发生了致命错误，则 CPU 在问题解决之前无法进入运行模式。

以下是一些致命错误的例子：

- 基本框架电源不正常
- 奇偶错误或 CPU 故障
- I/O 设置错误
- 某些程序错误

非致命错误

非致命错误是 CPU 标记出来的需要注意的错误。它们既不会禁止 CPU 从运行模式切换到编程模式，也不会阻止 CPU 进入运行模式。用户可以在编程时使用一些特殊继电器，使系统能够检测出这些非致命错误。然后有需要的话，您可以把系统安全关闭或切换到编程模式。

以下是一些非致命错误的例子：

- 电池电量低
- 所有的 I/O 模块错误
- 某些程序错误

查找诊断信息

诊断信息可以按消息细节的等级在许多地方查找到。

- CPU 会自动记录错误码和 FAULT 信息到两个独立的存储区域中。您可以通过手持编程器或 DirectSOFT 软件来查看。
- 手持编程器会显示错误码以及简短的描述。
- DirectSOFT 提供了错误码和一条错误信息
- 本手册的附录 B 中有一张完整的按错误码来排序的错误信息。

许多这些信息指向补充的存储区域。这些信息可以作为额外信息的参考。这些存储区域的参考是以 R 寄存器和特殊继电器的形式。以下两张表格给和某些类型的错误信息相关的存储器区域命名。特殊继电器表也包括可以在编程时使用的状态指示灯。有关每个特殊继电器的详细信息请参考附录 D。

与错误码相关的 R 存储器区域

错误等级	错误种类	诊断寄存器
用户定义	和 FAULT 指令一起使用的错误码	R7751
系统错误	致命错误码	R7755
	主要错误码	R7756
	细小错误码	R7757
语法	发生语法错误的地址	R7763
	语法检查时发现的错误码	R7764
CPU 扫描	上一次从编程模式切换到运行模式后的扫描次数	R7765
	当前扫描时间（ms）	R7775
	最小扫描时间（ms）	R7776
	最大扫描时间（ms）	R7777

与错误码相关的特殊继电器 (SP)

上电与实时继电器	
SP0	初始复位
SP1	常时 ON
SP3	1 分钟时钟脉冲
SP4	1 秒钟时钟脉冲
SP5	100ms 时钟脉冲
SP6	50ms 时钟脉冲
SP7	每隔一次扫描 ON
CPU 状态继电器	
SP11	强制 RUN 状态
SP12	TERM RUN 状态
SP13	TEST RUN 状态
SP14	TEST HOLD 状态
SP15	TEST PROGRAM 状态
SP16	TERMINAL PROGRAM 状态
SP20	STOP 继电器
SP21	BREAK 指令已执行
SP22	中断许可继电器
系统监控继电器	
SP40	重度异常继电器
SP41	轻度异常继电器
SP43	电池异常继电器
SP46	通讯错误继电器
SP47	I/O 配置异常继电器
SP50	FAULT 指令已执行
SP51	看门狗超时
SP52	语法错误
SP53	无法处理逻辑
SP54	智能模块通讯出错

累加器状态继电器	
SP60	小于标志继电器
SP61	等于标志继电器
SP62	大于标志继电器
SP63	零标志继电器
SP64	半借位标志继电器
SP65	借位标志继电器
SP66	半进位标志继电器
SP67	进位标志继电器
SP70	符号标志继电器
SP71	接点指定出错标志
SP73	溢出标志继电器
SP75	数据出错标志继电器
SP76	读零标志继电器
通讯监控继电器	
SP116	端口 2 通讯标志
SP117	端口 2 通讯错误标志
SP120	0 号槽模块忙
SP121	0 号槽通讯错误
SP122	1 号槽模块忙
SP123	1 号槽通讯错误
SP124	2 号槽模块忙
SP125	2 号槽通讯错误
SP126	3 号槽模块忙
SP127	3 号槽通讯错误
SP130	4 号槽模块忙
SP131	4 号槽通讯错误
SP132	5 号槽模块忙
SP133	5 号槽通讯错误
SP134	6 号槽模块忙
SP135	6 号槽通讯错误
SP136	7 号槽模块忙
SP137	7 号槽通讯错误

错误信息表

DL350 CPU 会自动记录系统错误码和您通过 FAULT 指令来创建的用户信息。CPU 会记录错误的代码及其发生的日期和时间。两个独立的区域存放着这些信息。

· 错误码表——系统可以记录最多 32 条错误。当错误发生时，以前记录的错误分别往下移，最近的错误放到顶部。当表格已放满时还发生错误，则排在最后一个错误将被丢掉。

· 信息表格——系统可以记录最多 16 条信息。当一条信息被触发，以前记录的信息分别往下移，最近的信息被放到顶部。当表格已放满时还发生错误，则排在最后一条消息将被丢掉。

以下表格说明了消息的错误表格。

日期	时间	消息
1993－05－26	08：41：51：11	*Conveyor-2 stopped
1993－04－30	17：01：11：56	*Conveyor-1 stopped
1993－04－30	17：01：11：12	*Limit SW1 failed
1993－04－28	03：25：14：31	*Saw Jam Detect

您可以通过 DirectSOFT 的 PLC Diagnostic 子菜单或手持编程器来访问错误码表格和消息表格。有关细节请查阅 DirectSOFT 和手持编程器的手册。

下面的例子说明了如何通过手持编程器的 5C 菜单来查看错误码。首先显示的是最近的错误码和消息。您可以通过 PREV 和 NXT 键来翻阅。

？？？？？？？？？？？？？？？？？？？？s-10hp 上无 5c 菜单？？？？？？？？

系统错误码

系统能够记录最多 32 个错误码。这些日志中的错误码是所有 DL305 系统生成的错误消息的子集。这些错误由 CPU 或手持编程器生成，这取决于实际的错误。附录 B 中有关于错误码的更详细的描述。

错误能在各种时候被检测到。尽管如此，绝大多数都是在系统上电，进入运行模式，或在手持编程器上键入信息时发生的错误或非法请求。

错误码	描述
E003	软件超时
E004	无效指令（RAM 奇偶错误）
E041	CPU 电池电量低
E043	存储卡电量低
E099	超出程序存储空间
E101	CPU 存储卡内容丢失
E104	写操作失败
E151	无效命令
E155	RAM 错误
E201	终端块丢失
E202	I/O 模块丢失
E203	保险丝断
E206	用户 24V 电源故障
E210	电源故障
E250	I/O 链通讯错误
E251	I/O 奇偶错误
E252	新的 I/O 配置
E262	I/O 超出范围
E312	通讯错误 2
E313	通讯错误 3
E316	通讯错误 6
E320	超时
E321	通讯错误
E499	打印指令中无效文本
E501	进入失败
E502	无效地址
E503	无效命令
E504	无效参数/数值
E505	无效指令
E506	无效操作

错误码	描述
E520	错误操作:CPU Run 模式
E521	错误操作:CPU Test Run 模式
E523	错误操作:CPU Test Program 模式
E524	错误操作:CPU Program 模式
E525	模式切换不在 TERM 模式
E526	离线单元
E527	在线单元
E528	CPU 模式
E540	CPU 被锁
E541	口令错误
E542	口令重置
E601	存储器满
E602	指令丢失
E604	参数丢失
E610	I/O 类型错误
E611	通讯 ID 错误
E620	超出存储器
E621	EEPROM 存储器非空
E622	无手持编程器 EEPROM
E624	只限 R 寄存器
E625	只限编程
E627	错误的写操作
E628	存储器类型错误（应为 EEPROM）
E640	比较错误
E650	手持编程器系统错误
E651	手持编程器 ROM 错误
E652	手持编程器 RAM 错误

编程错误码

下面的列表说明了编程时可能发生的错误。当您把 CPU 切换到 RUN 模式或使用手持编程器来检查程序时，这些错误会被检测到。同时 CPU 会把 SP52 置位并把错误码写入 R7755。附录 B 提供了有关错误码的更详细的信息。

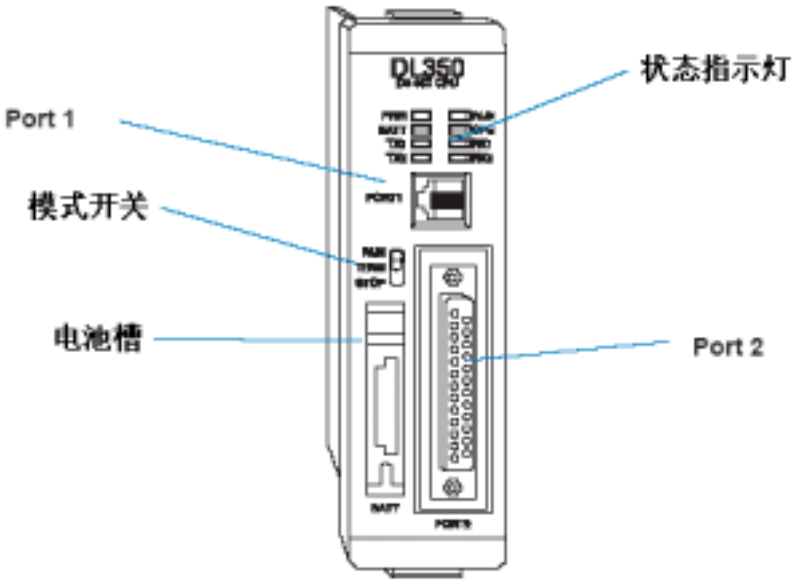
错误码	描述
E4**	CPU 中无程序
E401	缺少 END
E402	缺少 LBL
E403	缺少 RET
E404	缺少 FOR
E405	缺少 NEXT
E406	缺少 IRT
E412	SBR/LBL>64
E413	FOR/NEXT>64
E421	复制级参考
E422	复制 SBR/LBL 参考
E423	嵌套循环
E431	非法的 ISG/SG 地址
E432	非法的跳转（JUMP）地址
E433	非法的 SBR 地址
E434	非法的 RTC 地址
E435	非法的 RT 地址
E436	非法的 INT 地址
E437	非法的 IRTC 地址
E438	非法的 IRT 地址
E440	非法的数据地址
E441	ACON/NCON
E451	错误的 MLS/MLR
E452	输入线圈 I 用作输出线圈
E453	缺少 T/C
E454	错误 TMRA
E455	错误的 CNT
E456	错误的 SR

错误码	描述
E461	堆栈溢出
E462	堆栈下溢
E463	逻辑错误
E464	回路丢失
E471	复制线圈参考
E472	复制 TMR 参考
E473	复制 CNT 参考
E480	CV 位置错误
E481	CV 未连接
E482	CV 超出
E483	CVJMP 位置错误
E484	无 CV
E485	无 CVJMP
E486	BCALL 位置错误
E487	没有块被定义
E488	块位置错误
E489	CR 块定义错误
E490	没有块级
E491	ISG 位置错误
E492	BEND 位置错误
E493	BEND I 错误
E494	无 BEND

CPU 指示灯

DL350 CPU 的前部有一些指示灯，它们可以用来对系统进行诊断。以下的表格给出了对于不同的指示灯的状态可能出现的问题清单。其后有一个针对每个指示灯的详细分析。

指示灯状态	可能的问题
PWR（off）	1. 系统电压不正常 2. 电源/CPU 有故障 3. I/O 模块或其他组件的电源短路 4. 电功率消耗超出预算
RUN（亮不了）	1. CPU 编程错误 2. 开关在 TERM 位置 3. 开关在 STOP 位置
CPU（on）	1. 电气干扰 2. CPU 有缺陷
BATT（on）	1. CPU 电池电量低 2. 缺少 CPU 电池或接线不良
TX1	1．Port1 发送数据
RX1	1．Port1 接收数据
TX2	1．Port2 发送数据
RX2	1．Port2 接收数据



PWR 指示灯

CPU 的电源指示灯的不亮一般有 4 种情况：

1. 基本框架所使用的电源不正确或无法使用
2. 框架电源故障
3. 其他系统组件发生短路
4. 系统功率消耗超过预算

框架使用的电源不正常

如果框架使用的电源不正常，则 CPU 和（或）基本框架也许不能正常工作或根本无法工作。

警告：为了避免触电事故的发生，检查接线时一定要给系统断电。

1. 首先，切断系统的电源来检查所有的进线。
2. 如果您使用的是一个独立的中断面板，请检查那里的接线是否接错。
3. 如果检查无误，请给系统上电并测量框架上的电源端子，检查输入电压是否符合要求。如果电压不正常，请关闭系统并设法去解决问题。
4. 如果所有的接线和输入电源都正常，则请把框架寄给我们来维修。

CPU 故障

最好的检查 CPU 是否有故障的方法就是换一块新的 CPU 来试试。如果您发现主电源波动比较大，则 CPU 和基本框架已损坏。如果您怀疑这是电源损坏的原因，您可以在将来安装一个可以去除有害电压毛刺的整流器（line conditioner）。

装置或模块导致的电源不工作

如果使用系统的 5V 电源的模块或外部设备发生了故障，则有可能导致电源不能正常工作。该 5V 是来自框架上的端子或通讯端口。

测试此类设备的方法：

1. 去掉 CPU 的电源。
2. 断开所有的外部设备（也就是通讯电缆）和 CPU 断开。
3. 再次运行系统。

如果此时电源工作正常，则外部设备和电缆中至少有一个被短路了。如果此时电源还是工作不正常，则按照下面的步骤检查出问题的模块：

如果 PWR 指示灯亮，则问题可能出在某一个模块上。为了区分是哪个模块出了问题，切断系统的电源并每次拿走一个模块直到 PWR 灯亮为止。

请按照以下的程序：

· 切断框架的电源
· 拿走一个模块
· 重新给框架通电

弯曲了模块上的连接器针脚可能导致这种后果。检查连接部位是否有这种问题。

超出功率预算

如果在指示灯不亮以前，该机器已经在长时间内工正常好的话，一般不会是功率预算出了问题。超出功率预算的问题一般都是发生在上电的时候，因为此时 PLC 正处在工作状态而且输入点和输出点都需要更多的电流。

警告：如果超过了功率预算，PLC 会重启。如果您怀疑系统的功率预算有什么问题，请在此时仔细检查。超出了功率预算，将会产生一些导致人员伤害或设备损坏的不可预测的后果。请检验模块的功率损耗是否在所使用的框架的允许范围之内。详情请参考第四章系统设计和配置。

RUN 指示灯

如果 CPU 不能进入 RUN 模式（RUN 指示灯不亮），一般都是程序有问题，除非 CPU 发生了致命错误。如果发生了致命错误，CPU 灯会亮。您可以通过编程设备来纠正这个错误。
如果您在使用 DL350 并且试着用编程设备来切换模式，请确保开关是在 TERM 位置的。
对于这样的出错，手持编程器和 DirectSOFT 都会返回一条消息。您也可以通过 AUX 功能来检测出这个问题，这要取决于什么问题。大部分的常规编程出错是因为漏掉了“ END ”指令。所有的程序都要用“ END ”来结尾。附录 B 中有详细的错误码解释。

CPU 指示灯

如果 CPU 指示灯亮了，则 CPU 发生了一个致命错误。一般而言，这不是编程上的问题，而是硬件上的故障。您可以通过重新上电来解决这个问题。如果错误消除了，您应该监视系统并找出产生错误的原因。您会发现有时候问题是由于引入了外部电源的高频杂波而造成的。如果怀疑是接地问题，可以检查接地并安装电源滤波器。若重新上电后还是没有消除错误，或问题又产生了，您应该更换 CPU。

BATT 指示灯

如果 BATT 指示灯亮了，则说明 CPU 中的电池没有接好或需要更换。系统上电后会不断监视该电池的电压。

通讯问题

如果您不能和 CPU 进行通讯连接，请检查以下项目：
· 电缆没有连接好
· 电缆中有断线或内部接线有误
· 电缆没有正常中止或接地
· 与之相连的设备没有设定相应的波特率（上面的端口为 9600 波特，下面的端口可通过 56 菜单来设置）
· 与之相连的设备发送的数据有误
· 两者的地存在电位差
· 电源噪声产生连续的错误
· CPU 的端口坏了，需要更换
· 如果您使用的是 DirectSOFT，请参考快速手册中的故障排除一节
如果出现了错误，指示灯会一直亮直到通讯成功为止。

I/O 模块故障排除

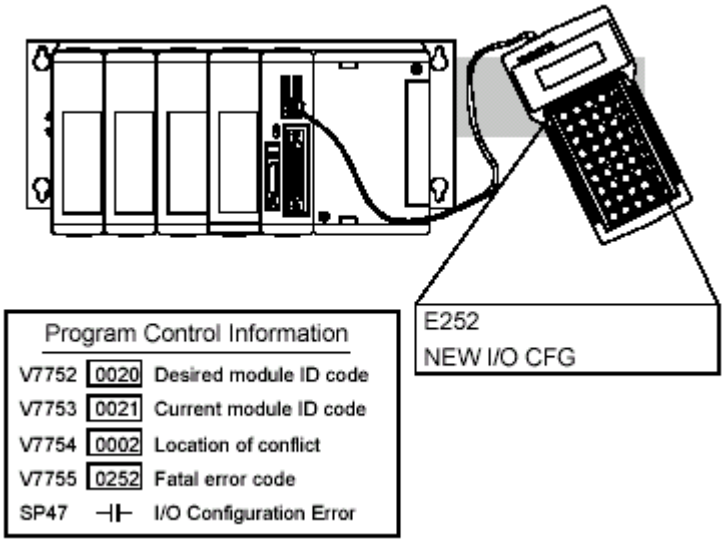
检查项目

如果您怀疑是 I/O 错误，可能有几种原因造成：

- 保险丝熔断
- 模块发生松动
- 24VDC 输出不正常
- 模块本身有故障
- I/O 配置检查检测到一个 I/O 配置的改动

I/O 诊断

如果模块没有提供一点问题的线索，请执行手持编程器上的 42 菜单或 DirectSOFT 中的 I/O diagnostics。这两种方法都能查看框架号，槽号及模块上的问题。 问题一旦解决，该灯就会灭掉。I/O 出错不会使 CPU 从 RUN 模式变为编程模式，但是有些特殊继电器可以用于梯形图程序来读取错误。应用程序就能采取相应的行动，比如进入编程模式或进行正常关闭。下图说明了出现错误的一个例子。



一些快速步骤

当处理 DL305 系列的 I/O 模块时，您必须知道一些情况。这些能够帮您更快地解决问题。

- 输出模块无法检测短路的或开路的输出点。如果您怀疑某个模块上的一个或几个输出点有问题，您应该测量公共端和该输出点之间的电压。请记住，当使用数字电压表时必须考虑到输出器件，比如双向可控硅或晶体管的漏电流。如果输出点没有接负载，则 OFF 的输出点有可能表现为 ON。
- 模块上的 I/O 指示灯只是逻辑上的指示灯。这意味着这些指示灯所表示的ON或OFF状态只是从CPU角度来说的。如果模块上的输出器件（双向可控硅或晶体管）被损坏了，则与之对应的指示灯依然能够正常工作。如果输入模块上的指示灯亮了，则说明输入回路是正常工作的。要确定是否工作正常，可以把输入信号去掉，再看看相应的指示灯是否灭掉。
- 当把现场设备接到 I/O 模块上时，漏电流的存在也许会成为一个问题。如果一个输出装置的漏电流过大，则有可能产生错误的输入信号。要解决这一问题，可以在输入回路中串一个电阻。阻值的大小取决于漏电流及电源电压的大小，但通常都是在 10K～20KΩ之内。同时要考虑到电阻的功率应符合要求。
- 最简单的判断一个模块是否有问题的方法就是用一块好的来代替它。尽管如此，如果您怀疑是其他的设备导致了模块的问题，则更换模块后应该也会发生同样的问题。需要注意的一点是，您最好在用其他模块替代之前先检查一下设备的接线和电源电压。

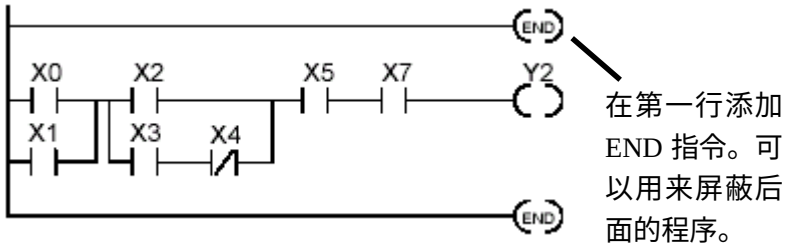
测试输出点

如果您要独立于程序外来检查 I/O 问题，请按以下的程序：

步骤	措施
1	使用手持编程器或 DirectSOFT 来连接 PLC。
2	切换到编程模式
3	找到地址 0
4	在地址 0 插入“ END ”指令。（这能够屏蔽用户的程序，从而防止 I/O 点的状态发生变化。）
5	切换到 RUN 模式
6	使用编程设备来测试相应的 I/O 点
7	当完成了 I/O 测试以后，删除地址 0 的“ END ”

警告：对某些系统应用而言，强制输出可能会产生一些不可预测机器动作，这有可能造成机器损坏或人员伤害。请确保您已对此作了充分的准备。

通过手持编程器按键来检测输出点



？？
？？

噪音故障排除

电气噪音问题

噪音问题是最难诊断的一个问题之一。电气噪音可以通过多种途径进入系统但可以归为两类：导电型的和辐射型的。判断噪音是如何进入系统有可能比较困难，但对两种噪音的处理方式却是类似的。

- 导电型的噪音是由于电线的接触，面板的接线等所产生的电气干扰。这些噪音有可能进入 I/O 模块，电源接线，通讯地，或底盘接地。
- 辐射型噪音是不通过直接的电气连接而是类似于无线电波的形式的电气干扰。

降低电气噪音

当电气干扰不能被消除时，我们可以降低它的程度使之不会影响系统的正常工作。

- 大多数的噪音问题是系统接地不良引起的。有一个良好的接地点是唯一的也是最好的解决噪音问题的办法。若接地点是无效的，那么就在尽可能靠近系统的地方安装一根接地棒。保证所有的接地线都是单点接地的并且不以菊花链的方式进行互连。用接地的金属来包围整个系统。在把噪音引入系统时，一根不固定的接线所产生的影响相当于一根大的天线；所以，您应该接好系统的每一根线。不固定的接地线比系统中的其他接线更容易产生问题。如果针对如何来给系统接地还有什么问题的话，您可以参考第二章中的相关内容。
- 电气噪音还有可能通过 CPU 和 I/O 模块的电源进入系统。安装一个隔离变压器可以解决这一问题。DC 电源应该有一个良好的接地。切换直流电源一般会产生比线性电源更多的噪音。
- 把输入接线和输出接线分开来。绝对不要把 I/O 接线靠近高电压的线路。

启动机器和调试程序

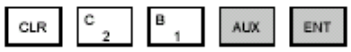
DL350 CPU 提供了一些功能可以帮助您在启动机器时或之前来调试程序。本节将对以下的主题进行讨论，这将会对您非常有帮助。

- 程序的语法检查
- 复制参数检查
- TEST 模式
- 特殊指令
- 运行时间编辑
- 强制 I/O

语法检查

尽管手持编程器和 DirectSOFT 都在写入程序时提供检查，但您有可能需要检查一个已修改的程序。两种编程设备都提供一种检查程序语法的方法。比如，您可以使用编程器上的 21 菜单，CHECK PROGRAM 或者 DirectSOFT 中的 PLC Diagnostics 菜单来检查程序的语法。这个功能能够检查出程序中的多种错误。以下例子说明了如何使用手持编程器上的这一功能。

使用 21 菜单来进行程序检查




```
AUX 21 CHECK PRO
1:SYN 2:DUP REF
```

选择语法检查（默认选择）

（若程序很短则不会显示“BUSY”）

```
BUSY
```

将会显示两者之一

错误显示（举例）

```
$00050     E401
MISSING END
```

语法无误显示

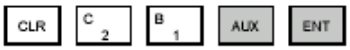
```
NO SYNTAX ERROR
?
```

附录 B 中有完整的错误码。如果您检查出一个错误，按 CLR 键就会显示发生错误的指令。纠正错误以后再次检查语法，直到显示“ NO SYNTAX ERROR ”。

检查重复线圈号

您可以查看同一线圈的重复使用。两种编程设备都有这个功能。例如，您可以通过使用编程器的 21 菜单，CHECK PROGRAM 或 DirectSOFT 中的 PLC Diagnostics 菜单选项来检查重复使用的线圈。下面的例子说明了如何用手持编程器来操作。

使用 21 菜单来进行程序检查




```
AUX 21 CHECK PRO
1:SYN 2:DUP REF
```

选择检查重复线圈

(You may not get the busy display if the program is not very long.)

```
BUSY
```

将会显示两者之一

错误显示（举例）

```
$00024  E471
DUP COIL REF
```

语法无误显示

```
NO DUP REFS
?
```

如果您检查到一个错误，按 CLR 键会显示发生线圈重复的指令。纠正问题后继续执行该操作直到没有发现重复线圈。

注意：您可以在不同的地方使用同一个线圈，尤其在使用级式编程和（或）OROUT 指令时。重复线圈号检查会找到这些输出点，尽管它们是被正确使用的。

TEST-PGR 与 TEST-RUN 模式

Test 模式允许 CPU 以 TEST-PGM 模式开始，进入 TEST-RUN 模式后运行若干个扫描周期后再回到 TEST-PGM 模式。扫描周期数可选择 1～65525。Test 模式同样允许您在 TEST-PGM 和 TEST-RUN 模式之间切换时保持输出点的状态。要切换到 Test 模式，用手持编程器可以按 MODE 键，用 DirectSOFT 可以选择 PLC MODES 菜单。使用 Test 模式的最基本的好处就在于可以在 CPU 回到 TEST-PGM 模式时保持某些输出点和参数的状态。而且切换到 TEST-PGM 模式后，CPU 可以保持定时器和计数器的当前值。

注意：只能通过 DirectSOFT 来设置扫描周期的数量。该功能在手持编程器上不支持。但是，您还是可以用手持编程器在 TEST-PGM 和 TEST-RUN 模式之间进行切换。

使用手持编程器，当您选择 TEST 模式后所进入的实际的模式取决与在这之前 CPU 处在何种模式下。如果 CPU 在 RUN 模式下，则切换成 TEST-RUN 模式。如果 CPU 在 PROGRAM 模式下，则切换成 TEST-PGM 模式。切换到 TEST 模式后，您就可以在 TEST-PGM 和 TEST-RUN 模式之间任意切换。DirectSOFT 的各种菜单选项提供了更加灵活的切换模式的方法。下面例子说明了如何来用手持编程器来选择 TEST 模式。

使用 MODE 键来选择 TEST 模式（例子假设为 RUN 模式下）

MODE NEXT ENT

```
*MODE CHANGE*
GO TO T-RUN MODE
```

按下 ENT 键来选择 TEST-RUN 模式

ENT

```
*MODE CHANGE*
CPU T-RUN
```

再次使用 MODE 键，可以回到 RUN 模式，进入 PROGRAM 模式，或进入 TEST-PGM 模式。

CLR MODE NEXT NEXT ENT

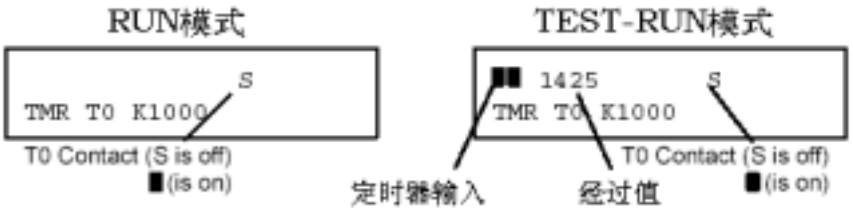
```
*MODE CHANGE*
GO TO T-PGM MODE
```

按下 ENT 来选择 TEST-PGM 模式

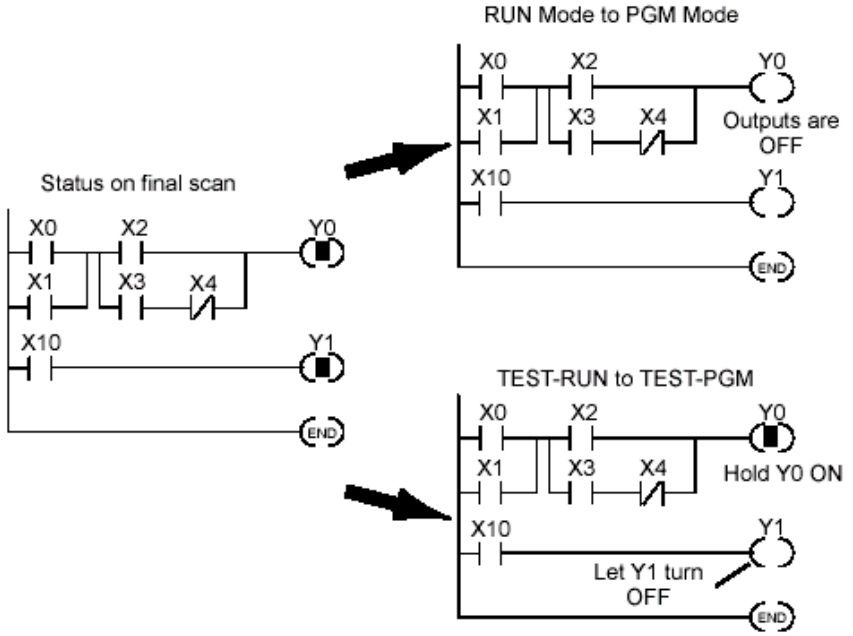
ENT

```
*MODE CHANGE*
CPU T-PGM
```

TEST 显示：在手持编程器上使用 TEST 模式时，您还可以查看到一些细节信息。对于一些指令，TEST-RUN 模式下看到的信息可能比 RUN 模式下看到的更详细。下面说明了定时器指令在两种模式下的不同显示。



输出保持：输出保持功能是非常有用的，因为它允许您维持一些关键的 I/O 点的状态。有时您需要修改程序，但又不希望某些操作停止。当在正常的 RUN 模式下，切换到编程模式时输出点都会 OFF。在 TEST-RUN 模式下，您可以设置每个输出点的状态或者让它保持先前的状态。该功能可以在 DirectSOFT 中设置。下图说明了 RUN 和 TEST-RUN 模式之间的不同。



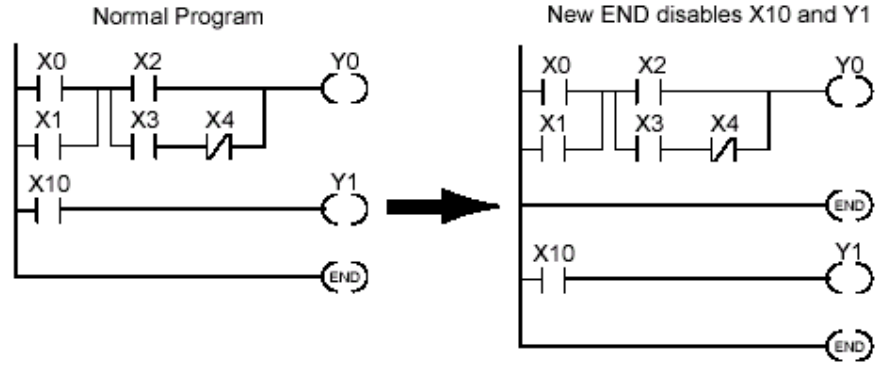
在您认为 TEST 模式是最好的选择之前，DL350 CPU 同样可以在 RUN 模式下修改程序。TEST 模式和 RUN TIME EDIT 功能的主要区别在于您不必逐个设定需要输出保持的输出点。当您使用 RUN TIME EDIT 功能时，CPU 在程序更新时会自动保持所有输出的当前状态。

特殊指令

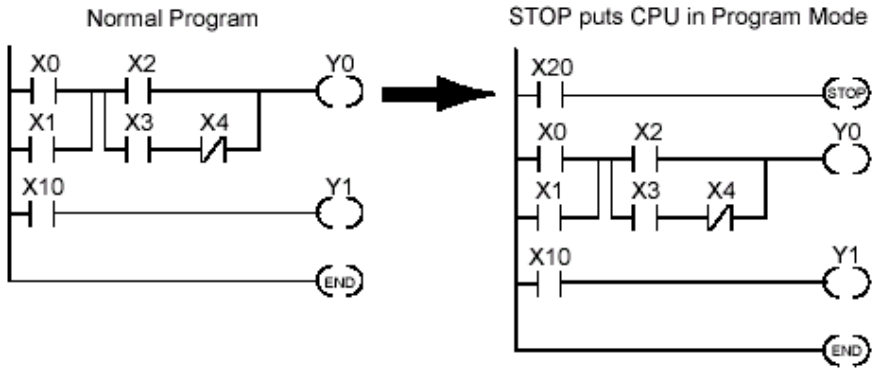
有一些特殊的指令有助于您在启动机器时顺利调试程序。

- END
- PAUSE
- STOP

END 指令 如果您需要快速地屏蔽程序，在不需要的程序处添加一条 END 指令。当 CPU 扫描到 END 指令，它将认为这是程序的末尾。下图说明了这个例子。



STOP 指令：在启动机器时，有时您需要快速关闭所有的输出点并回到编程模式。除了使用 TEST 模式，您还可以使用 STOP 指令。当执行该指令后，CPU 自动推出 RUN 模式并进入 PROGRAM 模式。记住，在 PROGRAM 模式下所有的输出点都 OFF。以下例子说明了使用一个条件来使 CPU 进入 PROGRAM 模式。



在上面的例子中，如果给 X20 一个触发信号，CPU 就会进入 PROGRAM 模式并关闭所有的输出点。

运行期间编程（RUN TIME EDITS）

DL350 CPU 允许您在运行时修改程序。这些修改不是“ bumpless ”。相反，CPU 扫描不断地被中断（输出点维持原来的状态）直到程序完全修改完。这意味着如果输出点是 OFF 的，它会仍然保持 OFF 直到完成了程序的修改。同理，如果原来是 ON 的，它也会保持 ON。

警告：只有经授权的人员在对系统的每一个方面都非常了解的时候才能修改程序。RUN 模式下的修改会立即生效。请保证您已经彻底考虑了由此带来的后果并设法使人员伤害或机器损坏的可能性最小化。在运行期间编程的时候，一些重要的操作顺序会改变。

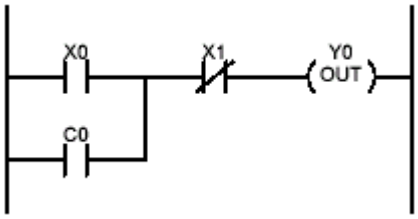
1. 如果新输入的指令中存在语法错误，CPU 将无法进入 RUN 模式。
2. 如果您删除的输出线圈当时为 ON，则该输出点的状态会一直保持直到您用编程设备来强制它为 OFF。
3. 在运行期间编程时输入点的状态不是得到普遍承认的。所以，如果您正在使用一个高速计数操作并且一个重要的输入点为 ON，则 CPU 有可能看不到这个变化。

并不是所有的指令都能用于运行期间编程。以下列举了能够的指令。

Mnemonic	Description
TMR	Timer
TMRF	Fast timer
TMRA	Accumulating timer
TMRAF	Accumulating fast timer
CNT	Counter
UDC	Up / Down counter
SGCNT	Stage counter
STR, STRN	Store, Store not
AND, ANDN	And, And not
OR, ORN	Or, Or not
STRE, STRNE	Store equal, Store not equal
ANDE, ANDNE	And equal, And not equal
ORE, ORNE	Or equal, Or not equal
STR, STRN	Store greater than or equal Store less than
AND, ANDN	And greater than or equal And less than

Mnemonic	Description
OR, ORN	Or greater than or equal Or less than
LD	Load data (constant)
LDD	Load data double (constant)
ADDD	Add data double (constant)
SUBD	Subtract data double (constant)
MUL	Multiply (constant)
DIV	Divide (constant)
CMPD	Compare accumulator (constant)
ANDD	And accumulator (constant)
ORD	Or accumulator (constant)
XORD	Exclusive or accumulator (constant)
LDF	Load discrete points to accumulator
OUTF	Output accumulator to discrete points
SHFR	Shift accumulator right
SHFL	Shift accumulator left
NCON	Numeric constant

使用程序的逻辑表示来描述该过程是如何工作的。在右边的例子中要把 X0 改成 C10。注意，现在假设 CPU 在 RUN 模式下。



使用 MODE 键切换到运行期间编程。

MODE NEXT NEXT ENT

MODE CHANGE
RUN TIME EDIT?

按下 ENT 键来选择运行期间编程

ENT

MODE CHANGE
RUNTIME EDITS

找到您所要修改的地方（X0）

SHIFT X SET A 0 SHIFT FD REF FIND

$00000 STR X0

按下方向键到 X 然后键入新的接点（C10）

→ → SHIFT C 2 B 1 A 0 ENT

RUNTIME EDIT?
STR C10

按 ENT 键来确定

ENT

OR C0

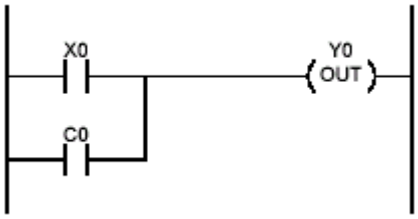
强制 I/O

有些时候，尤其是在机器启动和排除故障的时候，您需要强制功能来使一个 I/O 点为 ON 或 OFF。在您用编程设备的强制功能之前，应该了解 DL350 是如何处理强制请求的。

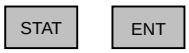
警告：只有经授权的人员在对系统的每一个方面都非常了解的时候才能修改程序。RUN 模式下的修改会立即生效。请保证您已经彻底考虑了由此带来的后果并设法使人员伤害或机器损坏的可能性最小化。

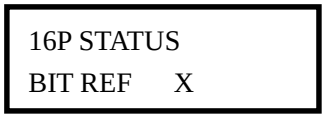
· **常规强制——**这类强制可以临时修改某个离散位的状态。例如，也许您需要使一个输入点 ON,尽管它是 OFF 的。您可以修改存放在镜像寄存器中的某个点的状态。在下一个扫描周期写入寄存器值时，该赋值才有效。这在测试情况下是最有用的，因为您也许需要强制某个位来触发一个动作。

下面举了一个简单的例子来说明如何使用手持编程器来强制 I/O 点。镜像寄存器的内容不随输入模块上的输入点状态的变化而变化。而且，程序的运行结果也不会刷新输出点的镜像寄存器。现在假设 CPU 在 RUN 模式下。

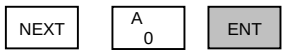


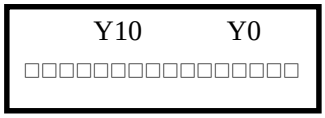
屏幕无任何显示的时候，使用以下的按键



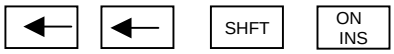


使用 PREV 和 NEXT 键来选择 Y 数据类型。（Y 出现后，按 0 使之从 Y0 开始。）

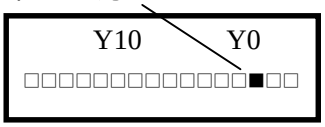




使用方向键来选择某个点，然后使用 ON 或 OFF 来改变它的状态



现在 Y2 为 ON



采用直接访问的常规强制

在屏幕无任何显示的时候，用以下的方式来给 Y10 为 ON。

SHFT | Y MLS | B 1 | A 0 | SHFT | ON INS

BIT FORCE
Y10 ■

在屏幕无任何显示的时候，用以下的方式来给 Y10 为 OFF。

SHFT | Y MLS | B 1 | A 0 | SHFT | OFF DEL

BIT FORCE
Y10 □